Matrix Solvers

Prof. Somnath Roy
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

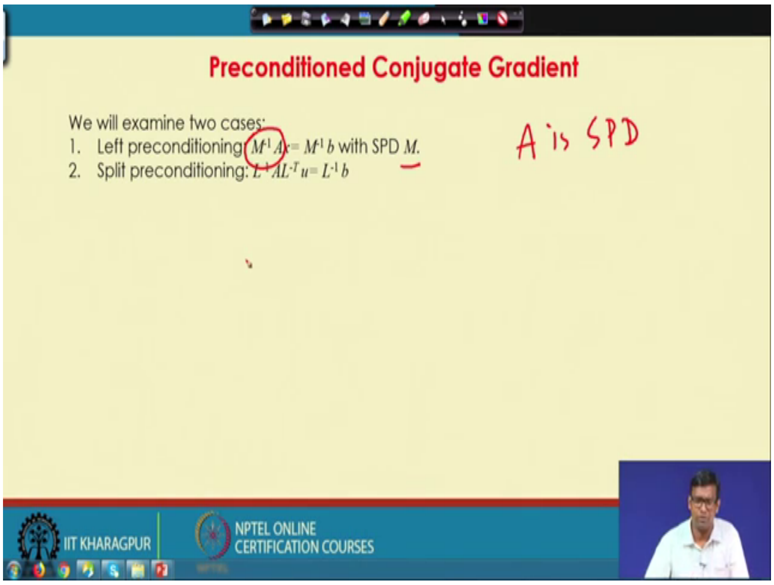
Lecture - 57
Preconditioned Conjugate Gradient

Welcome. So, we will looking into Preconditioning Techniques and we have discussed about left right and split preconditioning and this session we look into how preconditioning techniques can be applied over conjugate gradient method or how conjugate gradient method can be modified if we are using a pre conditioner.

The first example of preconditioned application of the conditioner we considered; considering to be conjugate gradient method, because conjugate gradient is only applicable for symmetric positive definite matrices. And therefore, we have to see that whether we can choose proper pre conditioner so, that symmetric positive definiteness is being maintained, and then how the resultant application comes out to be.

Pre conditioners can also be used over other solve solution techniques; however, because the fact that Krylov space based solvers are faster solvers. So, we will only discussing preconditioner over Krylov subspace based solvers, because these are they are already faster solvers and we need to make them more fast. For steepest descent or for Gauss Seidel we are not discussing preconditioning techniques because instead of doing that rather we will use Krylov space based solvers to get faster solvers solutions. But when Krylov space based solvers are restricted due to the poor conditioning number of the matrix we should think of using preconditioner so that we can get faster solution on top of that so, that we are not limited by the nature of the matrix; well.

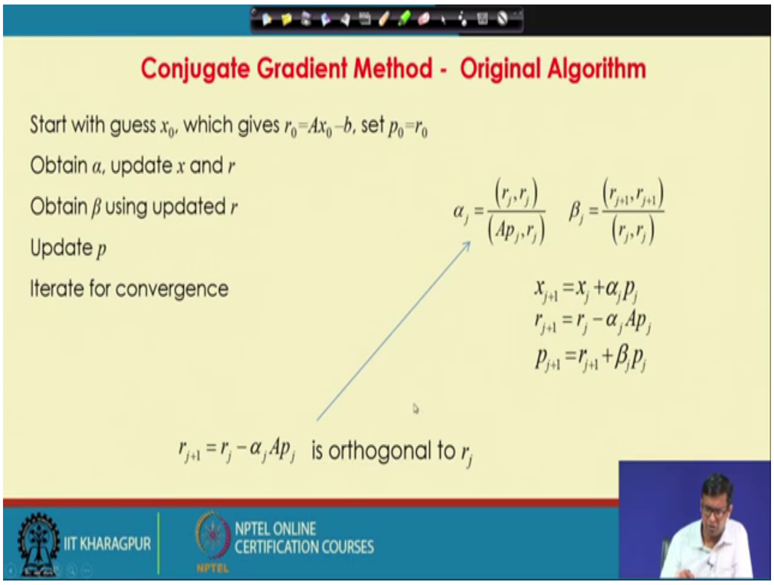
(Refer Slide Time: 01:56)



So, we will discuss two cases left preconditioning which is M inverse Ax is equal to M inverse b choosing symmetric positive definite M so, that the this matrix remains symmetric sorry sorry. So, that this matrix M inverse A this matrix remain symmetric positive definite. And conjugate gradient is only applicable when A is also a symmetric positive definite matrix.

So, if M is symmetric positive definite, M inverse A will remain symmetric positive definite and L inverse a L in L inverse AL inverse transpose will also be a systematic positive definite matrix. So, conjugate gradient will be applicable here both these equations can be solved using conjugate gradient. Now, we will see how what is the implementation of that.

(Refer Slide Time: 02:46)



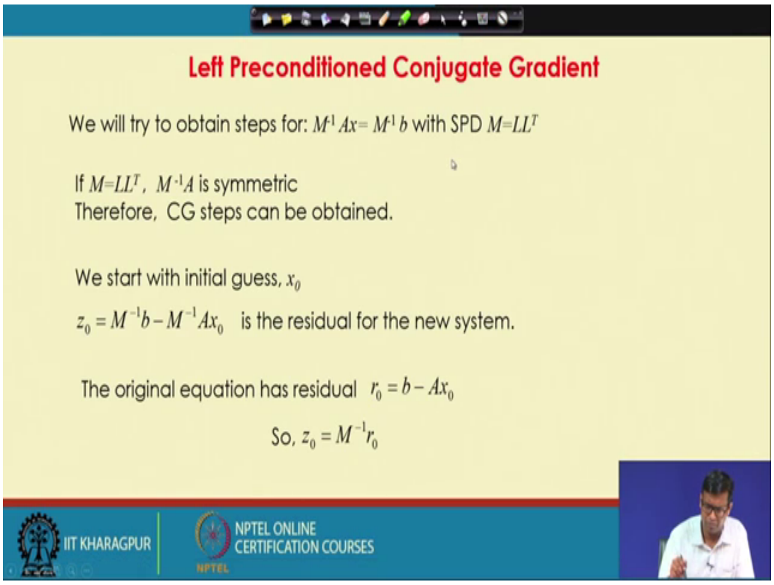
So, original conjugate gradient method without any preconditioning it started. So, it starts with a guess x 0 and computes the initial residual r 0 is equal to A x naught minus b and sets p naught is equal to r naught and then obtain alpha and update x and r alpha is obtained using the a conjugate c of p vectors and orthogonalization of r vectors which is r j transpose rj by Ap j r j transpose Ap j is alpha j similarly, we obtain another parameter beta j based on which p will be updated.

So, obtained alpha you and update x and r. So, one alpha is obtained x and r is updated, and now use the updated r to obtain beta and once beta is obtained, update the auxiliary vector p and then iterate for converges; that means, do these steps again and again till you get a converge solution where the residual is a very small value; the mod of the L two norm of this vector is a very small r L two norm of r is very small.

So, now we will see how, if we consider a preconditioned system, how these equations will be modified One idea can be that you explicitly compute an M inverse multiply that with a, get M inverse a multiply that with get M inverse b and solve it. But explicitly computing M inverse might be of problem at different stages because we are thinking of inverting a matrix or we are thinking of getting LU transformation of one particular matrix and then getting inversion and then doing a matrix multiplication.

So, this might be of more complication rather we will start with a matrix which is a form M inverse A, and we will assume that through some method we are getting the M inverse a from how we will discuss it later. And then we apply conjugate gradient method over that M inverse a matrix. However, we will keep in mind that our original Krylov subspaces based on r 0 a r 0 s square r 0 etcetera. So, while calculating r we use the property that r j plus 1 r j minus a alpha j Ap j is orthogonal to r.

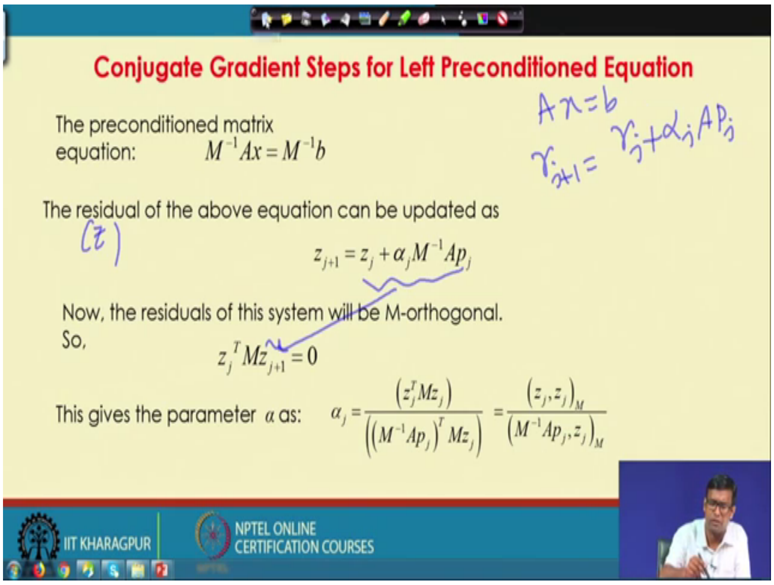
(Refer Slide Time: 05:15)



So, left preconditioning of conjugate gradient, will try to obtain steps for M inverse Ax is equal to M inverse A with M being symmetric positive definite, LL transpose. If M is LL transpose M inverse A is also symmetric therefore, conjugate gradient can be obtained. We start with an initial guess x 0; z 0 is a new residual which is M inverse b minus M inverse A x 0 because the new equation system is M inverse A x minus M inverse b the new residual is M inverse Ax 0 subtracted from M inverse b it is the new residual.

The original residual equation has residual r naught is equal to b minus Ax naught. The relation between the original residual and the new residual can be obtained as z naught is equal to M inverse r naught. So, now the if we are thinking of solving this equation,we have to go with a new residual which is z naught z or z naught.

(Refer Slide Time: 06:22)

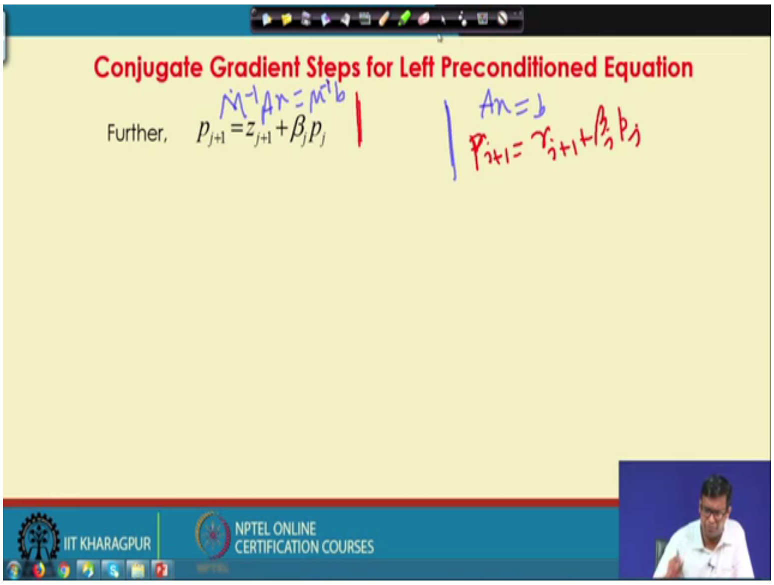


The precondition matrix equation is M inverse Ax is equal to M inverse b, the residual of the above equation can be updated as very similarly M inverse A is a coefficient matrix now. So, if we can remember our older update was r j plus 1 is equal to alpha j is equal to r j sorry r j plus 1 is equal to r j plus alpha j Ap j, that is our older update for Ax is equal to b equation.

Now, we have an equation M inverse Ax is equal to M inverse b, and the residual is z. So, z can be updated it is a very similarly as z j plus alpha j M inverse Ap j. Now, r j and r was orthogonal as we are discussing M inverse when we discussing a matrix, now when we will discuss a M inverse a matrix the new residual z j plus 1 and z j will be M orthogonal; that means, z j plus 1 z j transpose M z j plus 1 is equal to 0. So, dot product of z j with M z plus j plus 1 is equal to 0.

That gives the parameter alpha as alpha j is z j transpose M z j divided by if we substitute this here, if you substitute the z j plus 1 here we get an alpha j. So, we will substitute this as z j plus 1 and we will get an equation through which we can get alpha j is z j transpose M z j by M inverse Ap j transpose M z j. So, this is M product M dot product of z j z j z j transpose M z j and this will be M dot product of M inverse A p j and z j. So, we will see how to simplify these products.

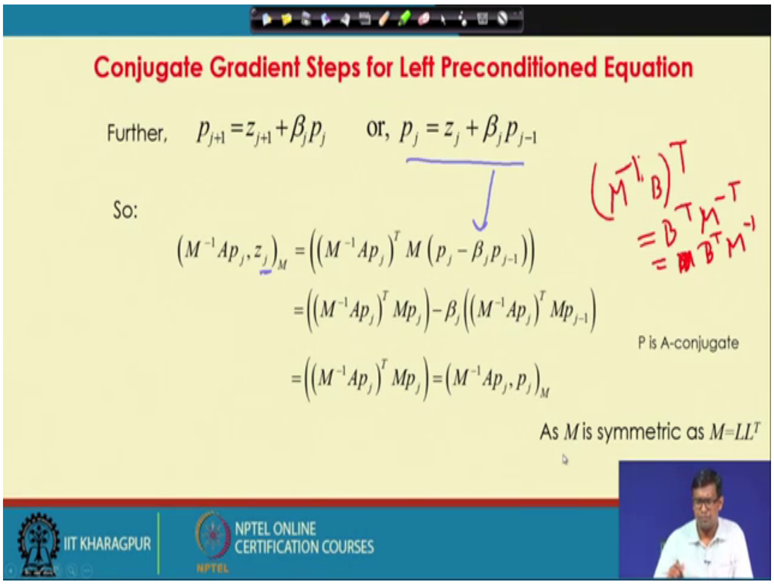
(Refer Slide Time: 08:31)



Further the auxiliary vector p can be updated this auxiliary vector is now, not auxiliary vector of x is equal to b rather auxiliary vector of M inverse Ax is equal to M inverse b. So, the update will have a relation with the new residual z earlier for Ax is equal to b. So, this is for M inverse x A x is equal to M inverse b this is for this equation. For Ax is equal to b the relation was something like r P j plus 1 is equal to rj plus 1 plus beta j p j.

So, for the M inverse Ax system this will be p j plus 1 is equal to z j plus 1 plus beta j pj M inverse. This is the relation for Ax is equal to b conjugate gradient therefore, this should be similarly we can find out the relation for p update of the auxiliary vector in conjugate gradient applied over M inverse A x is equal to M inverse M or p j is equal to z j plus beta j p j minus 1.

(Refer Slide Time: 09:49)



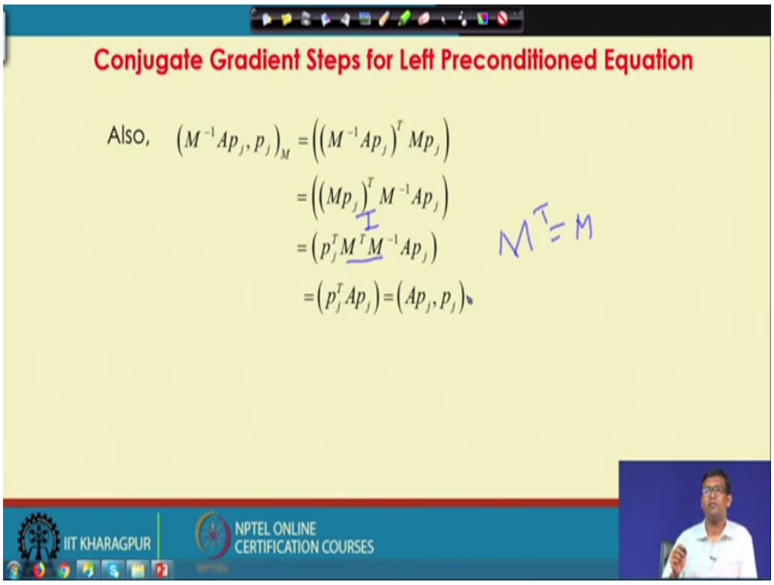
So, what we have already obtain the dot product between p j and z j now we will see how can we modify this. So, you got a relationship p j is equal to z j plus p j minus 1.

So, this the given dot product which is M inverse Ap j z j M is equal to M inverse A p j transpose M into z j which will be the z the term z j will be p j minus beta p j minus 1 and this is written from here. So, when we will do this we will have A p j transpose M inverse transpose and M inverse transpose is same as M inverse because M is a symmetric matrix. So, we will have a minus and that other part will be beta j M inverse p j transpose Mp j.

Now, if we. So, we break it down basically M inverse A p j Mp j transpose Mp j minus beta j which is a constant amount of it scalar and come out of it M inverse A p j transpose M inverse p j. And then we can write M inverse say B M inverse B transpose is equal to B transpose M inverse transpose and as M is a symmetric matrix. So, this will be B transpose M inverse.

So, this will be A p j transpose M inverse Mp j M inverse M will be pr identity matrix similarly this is A p j transpose M inverse M p j and p j and p j p j transpose A p j minus 1 will be a 0 vector because p is an a conjugate matrix p is a conjugate. So, we will see that this term will become 0 you can work it out this is just one step and this will be M inverse A p j transpose M p j or M inverse A p j p j of M as M is symmetric or M is LL transpose.

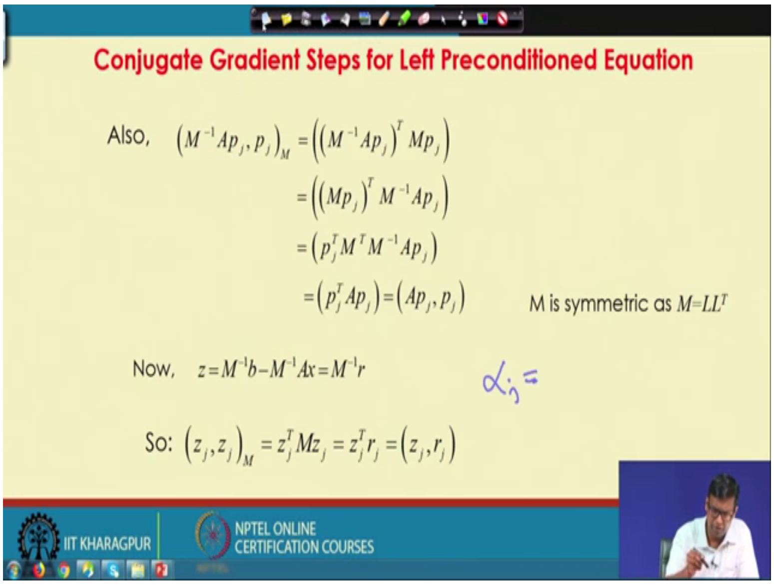
(Refer Slide Time: 12:39)



Further what we are discussing M inverse Ap j p j, M is M inverse A p j transpose m j which is Mp j transpose because we can change the order for in the dot product M inverse p j p j transpose M transpose M inverse A p j, and as M transpose M is symmetric. So, M transpose is equal to M. So, this will be an identity matrix and we will get p j transpose A p j or Ap j dot p j.

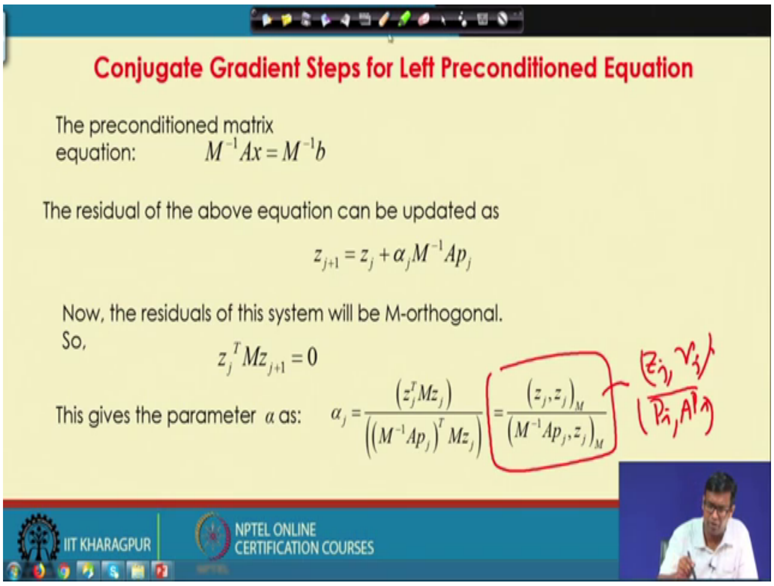
So, the denominator of the alpha calculation is simplified to Ap j transpose p j and remember this is the denominator what we obtained in conjugate gradient method also. However, the k v it is that an p j is calculated differently in this method. Because this is pj is an auxiliary vector of M inverse Ax is equal to M inverse b not for Ax is equal to b.

(Refer Slide Time: 13:50)



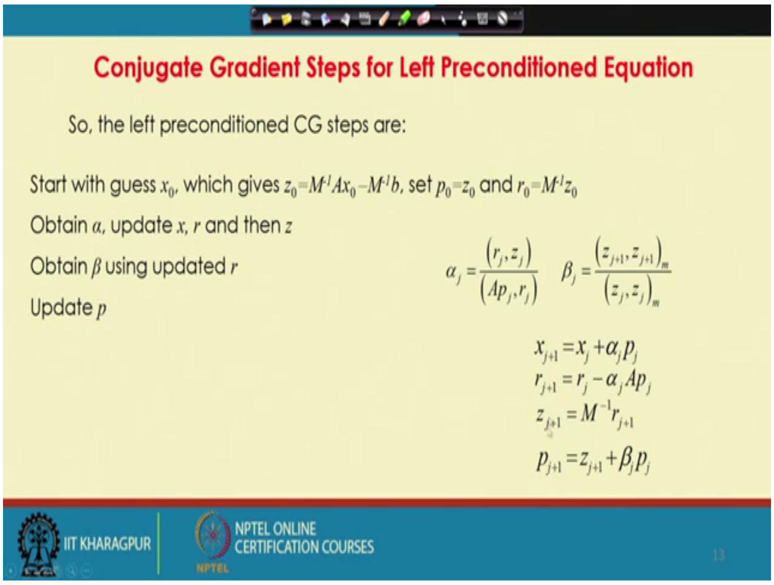
Now also z is equal to M inverse b minus M inverse Ax is equal to M inverse r. So, the M dot product of z j z j which is z j transpose Mz j can be written as z j transpose r r j or zj dot rj. So, my new alpha j which was equal to what was the initial form of alpha j we will again just once check.

(Refer Slide Time: 14:38)



The alpha j was, z j z j M by M inverse Ap j z j and this is now transferred to z j dot r j and this is transfer to P j dot A p j. This will be the new values the final form of alpha, which is a conjugate gradient solver algorithm will need to compute.

(Refer Slide Time: 15:20)



So, the left precondition conjugate gradient steps will be start with x 0, which will give z 0 is equal to guess x naught, which will give the new residuals z naught is equal to M inverse Ax naught minus M inverse b set p naught is equal to z naught and r r naught is equal to M inverse z naught.

Obtain alpha and then update x and r and then z. So, if we can update r you can update z or the same and obtain beta using the updated r. So, then the alpha is equal to r z z j by Ap j r j and you will obtain x j plus 1 is equal to x j plus alpha j p j. So, this also gives a direct way to update x instead of looking into z because x is (Refer Time: 16:11) x is the solution vector x is the solution vector. So, x will follow exactly same relation as it was following in the conjugate gradient method of x is equal to b. It will follow the same relation in the conjugate gradient method of M inverse Ax is equal to b because x remain same in the both in both the cases.

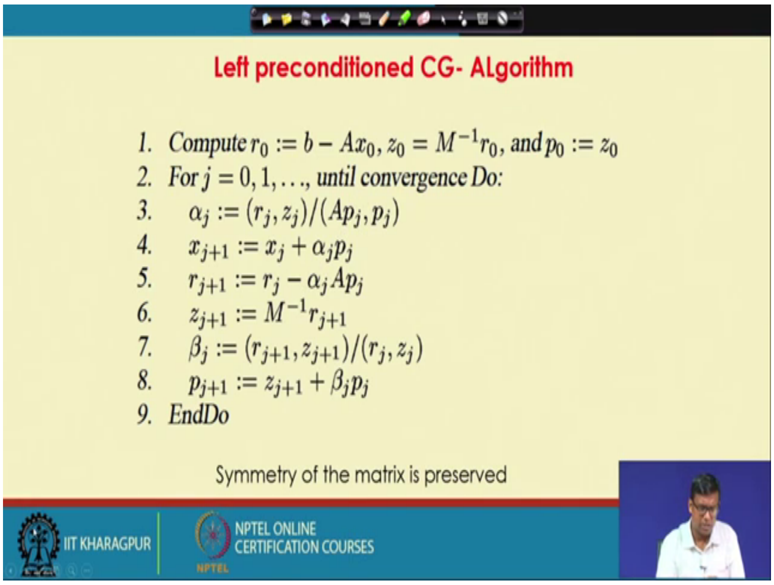
And rj plus 1. So, x j plus 1 is equal to x j plus alpha j p j similarly r j can be obtained as r j minus alpha j Ap j just substitute this into x is equal to b. X is equal to b equation is still holding and z j plus 1 is M inverse r j plus 1. So, again it becomes important here to compute M inverse; M inverse should be readily available with us, instead of actually looking into a LU factorization of a matrix and getting the lower transfer lower triangular form multiplying into upper triangular and inverse etcetera M inverse should be readily available that we need to check.

Later I will say that M how M is evaluated I will come into it later, which is also a very important part of this methods, but we will see that how them algorithms are modified provided we know to evaluate M and we are ready with M inverse or M inverse a in certain cases.

So, once z is updated you can update beta is j plus 1 z j plus 1, z j plus once in dot product by z j z j M dot product. So, earlier there are beta was dot product of the vectors now they are M products. And once beta is updated update p as p is equal to r rj plus 1 plus beta j p j p z j sorry it is not r now z, z p j plus 1 is z j plus 1 beta j p j. And then you can carry this p and r and z to get the new alpha new beta and continue on the iteration loop exactly the same way what you have down in conjugate gradient method, iterate for convergence.

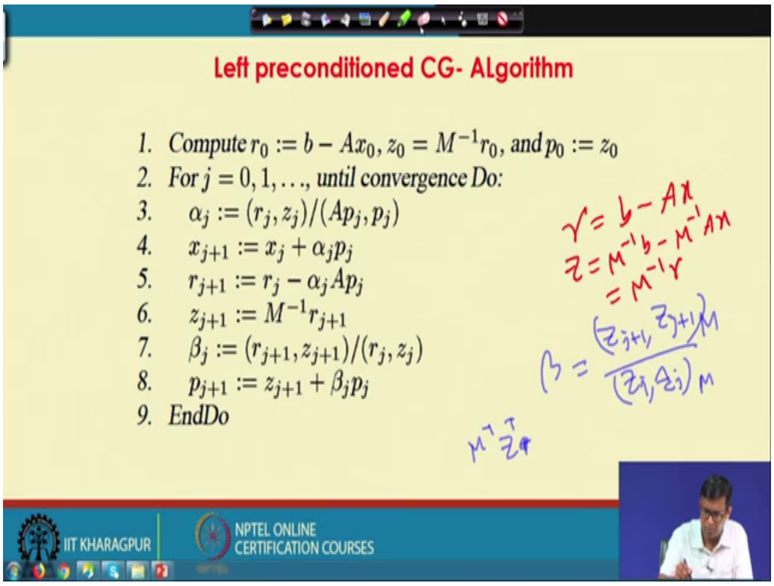
The conjugated ends will probably once look into the methods the conjugate gradient algorithm. Now can be formed using all these steps; starting with these steps start with these guess use all these steps and then do the iterations.

(Refer Slide Time: 18:42)



Which is compute r 0 is equal to b minus Ax 0 z 0 is M inverse r 0 p 0 is equal to z 0 for different iteration levels first compute alpha r j dot z j by Ap j dot p j. You know now in this calculation at least no M product is needed M is not needed here M was once needed at M inverse from here, but here M is not needed then x j plus 1 is equal to x j plus alpha j p j r j plus 1 is equal to r j minus alpha j Ap j z j plus 1. So, this and this if this holds because this is the exactor remains essentially same, this should hold then by b minus x we can get this relation this one once r is obtain z is equal to M inverse r.

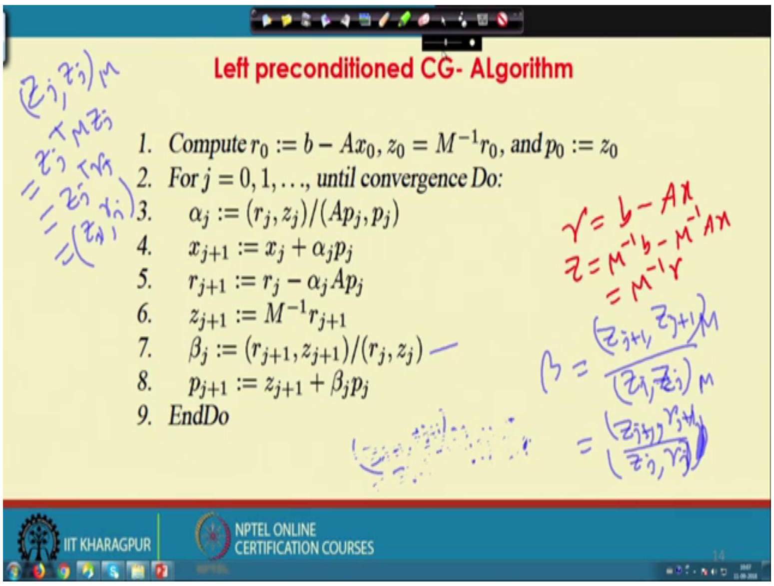
(Refer Slide Time: 19:33)



So, we should know this that r is equal to b minus A x z is equal to M inverse b minus M inverse Ax which is the precondition system sorry M inverse r.

So, then we get z is equal to M inverse r plus 1, beta is equal to r plus 1 z j plus 1 by r j z j which is earlier we have seen that beta is equal to z j plus 1 z j plus 1 M by z j z j M. Now, z transpose z transpose M transpose z sorry. So, what is z j plus 1?

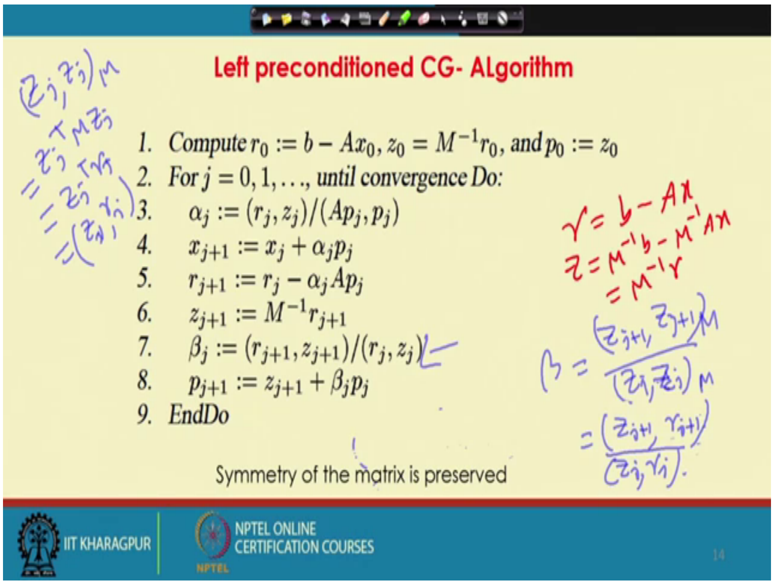
(Refer Slide Time: 20:28)



Z j plus 1 M this will be z j plus 1 transpose M z j plus 1 (Refer Time: 20:44) some way redo it we are trying to evaluate the beta which will be equal to same in a while we can check that that z j z j M, this product is equal to z j transpose M z j. And M z is equal to r. So, z j transpose r or z j rj z j dot r j.

So, this will be z j plus 1 dot r j plus 1 by z j dot r j this product will be this. So, we will get this particular beta evaluation from here. Z j plus 1 m z j plus 1 will be nothing, but dot product between z j plus 1 and r j plus 1 and if we write it for better clarity.

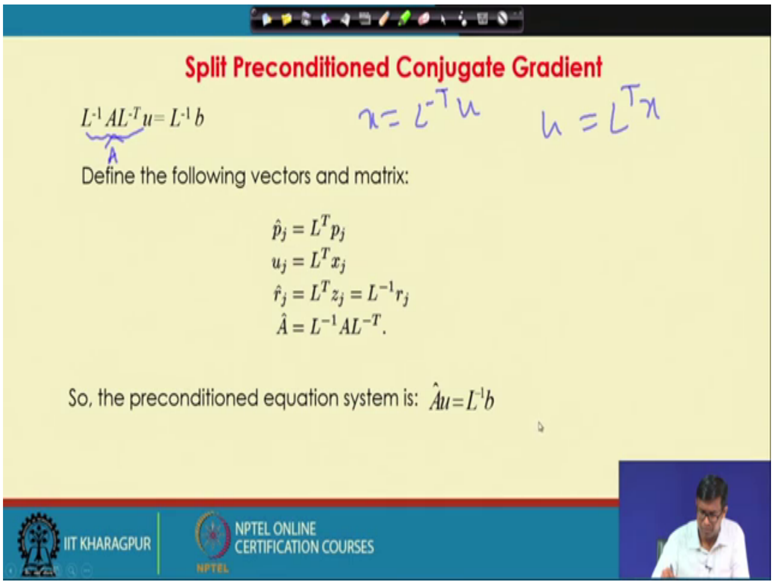
(Refer Slide Time: 22:38)



And this will be z j plus 1 dot r j plus 1 because z j z j in product is z j dot r and divided by z j dot r j, which is calculated here and we will get p j plus 1 is equal to z j plus 1 plus beta j p j the same way it should be (Refer Time: 22:58).

So, we get a; what we see that symmetricity of the matrixes preserved, that is how the M is chosen. So, that M inverse a is symmetric. And the solution is essentially very similar with the older conjugate gradient method; however, a new matrix inversion is added. This is a new matrix inversion which is added that i have to calculate M inverse and that is what we are telling that M this will be of further deliberation at the later stages that how M inverse should be obtained. So, that these steps do not add any extra overwrite here.

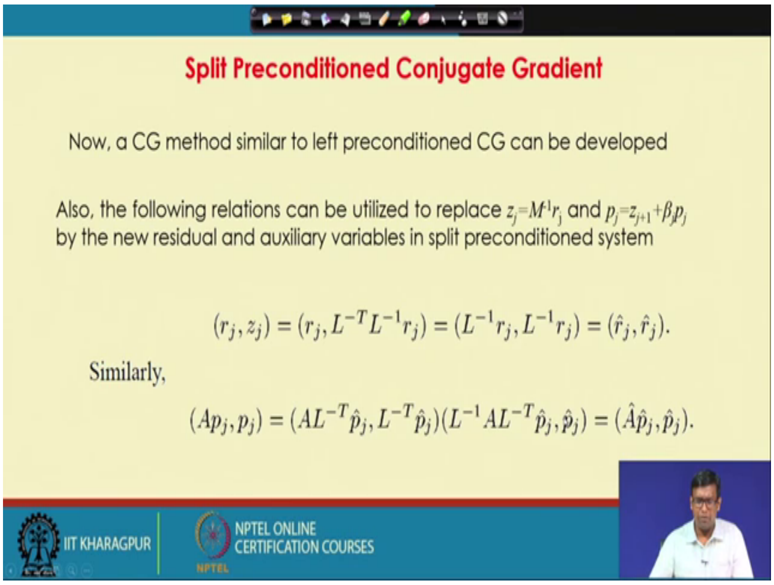
(Refer Slide Time: 23:57)



We go to the split precondition conjugate gradient which is L inverse a L in L inverse transpose u is equal to L inverse b, and this also needs solution of the equation that x is equal to L inverse transpose b. And, so you get the following vectors and matrices p hat is equal to L transpose p u is equal to L transpose x sorry x is equal to L inverse transpose u.

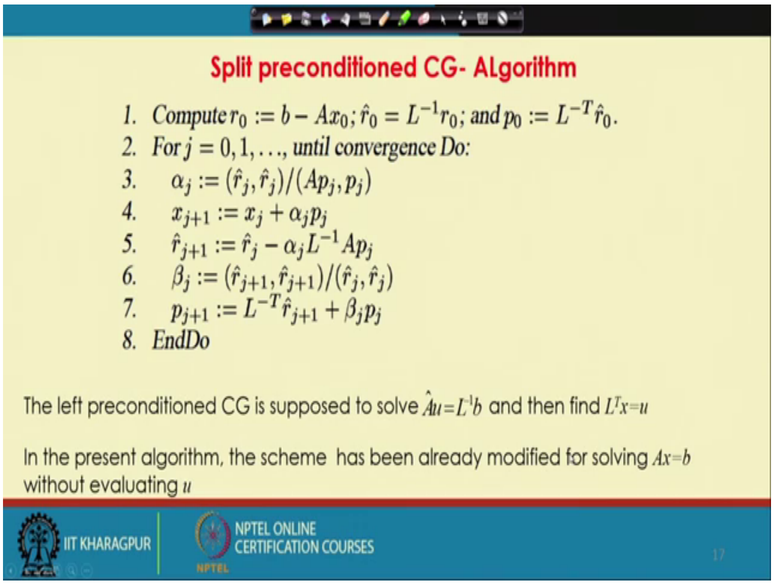
So, that will give me u is equal to L transpose x u is equal to L transpose x r hat is equal to L transpose j z j, which is L inverse rj because L yeah this will see later this comes from the previous discussion. So, r till hat is equal to L j z j; z j is the new vector and this is L inverse r j. And A hat is equal to L inverse AL inverse transpose. And the preconditioned equation system is A tilde u is equal to L inverse b. So, this becomes this equation system this becomes A hat; A hat u is equal to L inverse b.

(Refer Slide Time: 25:40)



Now, conjugate gradient method similar to the left preconditioned conjugate gradient can be developed. And this method there is the following relationships can be utilized which we have already seen with an older vector z j is equal to M inverse r j and p j is equal to z j plus 1 beta j p j by the new residual and auxiliary variables in the precondition systems. So, similarly we will get r j dot z j is equal to r hat j dot r hat j and A p j dot p j is equal to a hat p hat j dot p hat j.

(Refer Slide Time: 26:25)



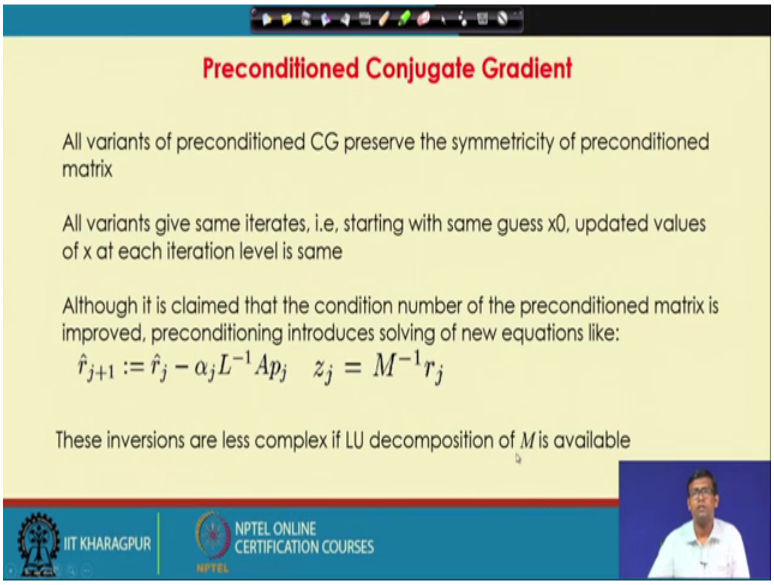
So, the method this is the derivation of the method is very similar to the left precondition method. Only thing we have a new equation L inverse transpose we are doing it in terms of you A hat u is equal to L inverse transpose b, and u is equal to L transpose u then equation system. So, we compute the initial residual r naught get L inverse transpose r naught is r naught hat obtain p naught is equal to L inverse transpose r naught.

And instead of directly write p p naught is equal to z naught this is the new form of p naught, then we obtained alpha we obtained update x update r based on x update. So, you can write b minus Ax is equal to 0 and get r from x and then update beta and get the p and do the iterations.

Then lift left precondition CG supposed to solve A tilde u is equal to L inverse b and then for find L transpose x is equal to u; however, in the present algorithm we are not explicitly evaluating u rather it is modified to solve Ax is equal to b. So, it is taken care of that both these sets of equations are solved together, by the way we are defining r tilde and p and then using this p to evaluate update x both this equation systems are taken care of.

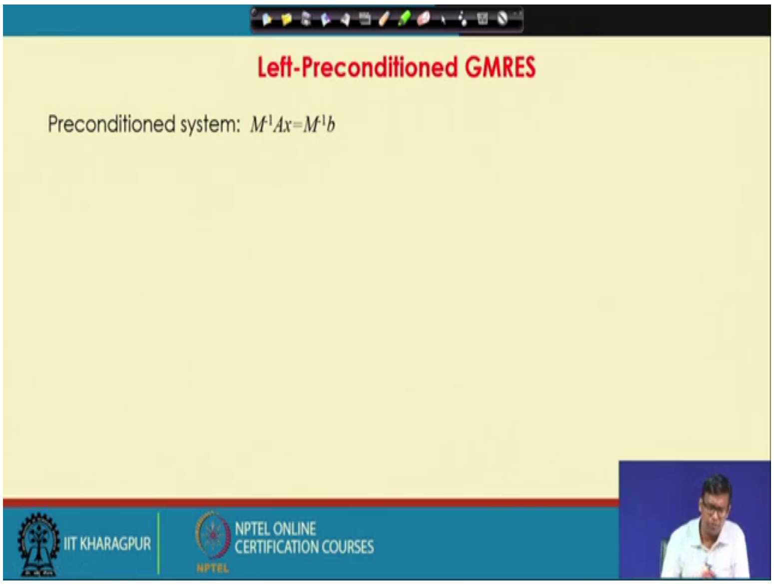
So, we are not solving it separately; however, something like L inverse transpose solution is required when we are finding out p and L inverse A this product is required when we are finding out r tilde. So, we are saying that any of the condition precondition system is adding some sort of overwrite in terms of solving a new equation system. So, this equation system the L inverse transpose or L inverse a this equation systems should be very; we will have should have much less complexity to solve you should have some way readily available solution in n steps, we should not encounter in cube number of steps some method we in which we can get n steps.

(Refer Slide Time: 28:50)



All variants of precondition CG preserve the symmetry of the precondition matrix all variants gave same iterates. So, we look in to split preconditioning we look in left preconditioning even if we think of right preconditioning, the iterate should be same that is starting with the values x 0 the updated value of x at each iteration level must be same. Although it is claimed that condition number of the precondition matrix in improve, the preconditioning increases new method introduces new equations like r tilde is equal to r theta L inverse A p j or z is equal to M inverse r j. So, the inverses are less complex; this inverse is less complex if LU decomposition is of M is already available, then the finding this inverse will be straightforward.

(Refer Slide Time: 29:39)



So, next session so, we have to see that how LU inverse of M is available, but before doing that in the next session we will look into how preconditioning of GMRES is obtained.

One group is for symmetric matrices we have looked into conjugate gradient method now for general matrices non symmetric matrices we will look into the GMRES method and how preconditioned algorithm for GMRES is available. And then we will see which is extremely important that how the inverse of the M matrix or how the actual precondition matrix and its application over the solution vectors can be are obtained so, that the computational complexities due to the preconditioning itself is not high, that add adds extra overwrite on the solution this methodology. So, we will also look into that in the next class.

Thank you.